Hướng dẫn sử dụng elasticsearch, kibana và logstash

# Elastichsearch.

Bắt đầu từ Elasticsearch 8.0, bảo mật được bật theo mặc định. Với tính năng bảo mật được bật, các tính năng bảo mật của Elastic Stack **yêu cầu mã hóa TLS**  cho lớp mạng truyền tải, nếu không **cluster** sẽ không khởi động được.

**Cấu hình 1 single-node Elasticsearch cluster:**

Nếu ta bắt đầu một single-node Elasticsearch cluster, **bảo mật** sẽ tự động được bật và cấu hình cho ta. Vào lần đầu tiên khởi động Elasticsearch, hệ thống sẽ tự động cấu hình cho ta, bao gồm:

* Certificates and keys được tạo **transport** và **HTTP layers.**
* **Transport Layer Security (TLS)** được tự động cấu hình và viết vào elasticsearch.yml.
* Mật khẩu được tạo cho user: elastic
* Một enrollment token được tạo cho Kibana.

Sau đó, ta có thể khởi động Kibana và nhập enrollment token, mã này có giá trị trong 30 phút. Mã này tự động áp dụng cài đặt bảo mật từ cluster Elasticsearch của ta, xác thực Elasticsearch với kibana\_system user, và viết cấu hình bảo mật vào kibana.yml.

Các lệnh sau đây bắt đầu một single-node Elasticsearch cluster cho phát triển (dev) hoặc testing:

1. Tạo 1 new docker netword for Elasticsearch và Kibana:
   * docker network create elastic
2. Khởi động Elasticsearch trong docker bằng lệnh sau, để khởi tạo mật khẩu cho tài khoản elastic và enrollment token cho Kibana.
   * docker run –name elastic –net elastic –p 9200:9200 –p 9300:9300 –it docker.elastic.co/elasticsearch/elasticsearch:8.4.2

Tại đây, nếu xảy ra lỗi: ERROR: [1] bootstrap checks failed. You must address the points described in the following [1] lines before starting Elasticsearch.

Đây là lỗi do, Docker được cấp bộ nhớ nhỏ hơn 4GB

Cách khắc phục:

Cmd: wsl –d docker-desktop 🡪 sysctl –w vm.max\_map\_count=262144

1. Nếu ta muốn cài đặt lại mật khẩu cho tài khoản “elastic” hoặc các tài khoản khác do ta lập ra:
   * docker exec –it elastic /usr/share/elasticsearch/bin/elasticsearch-reset-password.
2. Sao chép file **http\_ca.crt** từ docker sang máy chủ cục bộ của ta:
   * docker cp elastic:/usr/share/elasticsearch/config/certs/http\_ca.crt
3. Mở terminal để kết nối với Elasticsearch bằng các request xác thực, sử dụng tệp **http\_ca.crt**  mà ta đã sao chép từ vùng chứa docker:
   * curl –cacert http\_ca.crt –u elastic **https**://localhost:9200. – Lưu ý: phải là HTTPS chứ không phải HTTP.

Nếu tại đây xảy ra lỗi: curl: (60) schannel: CertGetCertificateChain trust error CERT\_TRUST\_REVOCATION\_STATUS\_UNKNOWN ??? – Hiện chưa có giải pháp ??? Nhưng vân có thể truy cập https:localhost:9200 bằng trình duyệt là được rồi. – có khả năng là do thiếu mật khẩu đăng nhập: curl –cacert http\_ca.crt –u elastic: $ELASTIC\_PASSWORD https://localhost:9200

Mã xác nhận (enrollment token) có hiệu lực trong 30 phút. Nếu ta cần khởi tạo một mã xác nhận mới hãy chạy câu lệnh sau:

* docker exec –it elastic /usr/share/elasticsearch/bin/elasticsearch-create-en rollment-token –s node (-s kibana để tạo mã xác nhận cho kibana)

Để các ứng dụng khác có thể truy cập vào địa chỉ của elasticsearch với https, trên máy tính, ta cần phải cài đặt tường lửa.

Ta có thể gửi dữ liệu vào Elasticsearch, bằng các gửi các đối tượng Json tới Elasticsearch qua HTTP – **Vậy có nghĩa là ta có thể truy cập Elasticsearch qua HTTP**.

* Cho dù đó là văn bản có cấu trúc hay không có cấu trúc, dữ liệu số, geospatial data 🡪 Elasticsearch sẽ lưu trữ và lập chỉ mục nó một cách hiệu quả theo cách hỗ trợ tìm kiếm nhanh.

(Hãy sử dụng Elastic Agent để thu thập dữ liệu từ các máy chủ hoặc vùng chứa mà ta cần theo dõi).

# Kibana

**Kibana:** Thay vì xây dựng và gửi yêu cầu REST trực tiếp đến Elasticsearch, ta có thể sử dụng **Discover** để tìm kiếm và lọc dữ liệu của mình, lấy thông tin về cấu trúc của các **Field** và hiển thị kết quả của ta.

Kibana yêu cầu một **data view** để truy cập dữ liệu Elasticsearch mà ta muốn khám phá. **Data view** chọn dữ liệu để sử dụng và cho phép ta xác định thuộc tính của các **Field**.

Một **data view** có thể trỏ đến một hoặc nhiều **indices**, **data streams** hoặc **index aliases**. Ví dụ, một data view có thể trỏ đến log data từ ngày hôm qua hoặc tất cả indices chứa trong data của ta.

Để sử dụng **Discover** và nhiều tính năng và công cụ khác có sẵn trong Elastic Stack, **dữ liệu của ta** phải được **liên kết**  với **data view.**

1. Chọn dữ liệu ta muốn làm việc với:
   1. Mở menu Kibana và chọn **Stack Management > Kibana > Data Views.**
   2. Chọn **Create data view**.
   3. Nhập tên phù hợp với một hoặc nhiều index Elastic search. Ta có thể tạo data view trên nhiều index bằng cách sử dụng ký tự đại diện \*. Ví dụ: b\*.
   4. Chọn **Create data view**.
2. Chọn **Discover** trên menu chính.
3. Chỉ đinh tiêu chí truy vấn bằng cách thêm bộ lọc:
   1. Chọn **Add filter.**
   2. Chọn một field hoặc 1 toán tử, nhập giá trị và nhấn Save.

**Trực quan hóa dữ liệu của ta.**

Ta có thể tạo hình ảnh trực quan và xây dưng trang tổng quan trong Kibana để hiểu dữ liệu và chia sẻ thông tin:

1. Mở menu chính của Kibana, sau đó chọn **Dashboard**.
2. Chọn **Create a Dashboard > Create visualization**.
3. Kéo và thả fields để tạo một visualization, sau đó chọn **Save and return**.

# Logstash

[2022-10-05T04:04:07,748][INFO ][logstash.agent ] Successfully started Logstash API endpoint {:port=>9600, :ssl\_enabled=>false}

[2022-10-05T04:04:08,024][INFO ][org.reflections.Reflections] Reflections took 56 ms to scan 1 urls, producing 125 keys and 434 values

[2022-10-05T04:04:08,279][INFO ][logstash.javapipeline ] Pipeline `main` is configured with `pipeline.ecs\_compatibility: v8` setting. All plugins in this pipeline will default to `ecs\_compatibility => v8` unless explicitly configured otherwise.

[2022-10-05T04:04:08,345][INFO ][logstash.javapipeline ][main] Starting pipeline {:pipeline\_id=>"main", "pipeline.workers"=>12, "pipeline.batch.size"=>125, "pipeline.batch.delay"=>50, "pipeline.max\_inflight"=>1500, "pipeline.sources"=>["/usr/share/logstash/pipeline/logstash.conf"], :thread=>"#<Thread:0x76bcdaa7 run>"}